

# Package ‘WordOfMouth’

January 20, 2025

**Type** Package

**Title** Estimates Economic Variables for Word-of-Mouth-Campaigns

**Version** 1.1.0

**Date** 2021-10-04

**Maintainer** Michael Scholz <michael.scholz@th-deg.de>

**Description**

Methods for estimating profit, profit-maximizing price, demand and consumer surplus of Word-of-Mouth-campaigns on mean-field networks.

**License** GPL-3

**Depends** R (>= 3.0.1), methods, stats, LambertW

**LazyLoad** yes

**ByteCompile** yes

**RoxygenNote** 7.1.2

**NeedsCompilation** no

**Author** Michael Scholz [cre, aut],  
Thomas Woehner [aut],  
Ralf Peters [ctb]

**Repository** CRAN

**Date/Publication** 2021-10-04 14:20:02 UTC

## Contents

WordOfMouth-package . . . . .	2
compareToFIMarket . . . . .	3
computeConsumerSurplus . . . . .	4
computeDemand . . . . .	5
computeInformationCostsThreshold . . . . .	6
computeOptimalPrice . . . . .	7
computeProfit . . . . .	8
computeRoundDemand . . . . .	9
computeWoMIntensity . . . . .	10

show, WoMCampaign-method . . . . .	10
show, WoMNetwork-method . . . . .	11
WoMCampaign-class . . . . .	12
WoMNetwork-class . . . . .	12
<b>Index</b>	<b>14</b>

---

WordOfMouth-package     *Estimates Economic Variables for Word-of-Mouth-Campaigns*

---

## Description

This packages provides classes, methods and functions for modeling Word-of-Mouth-campaigns. General model assumptions are:

- monopoly market
- no variable costs
- network is the mean-field case of percolation
- only those persons who bought a product will forward information about it

## Details

Package: WordOfMouth  
 Type: Package  
 Version: 1.1.0  
 Date: 2021-10-04  
 License: GPL-3  
 Depends: R (>= 3.0.1), methods

## Author(s)

Michael Scholz <michael.scholz@th-deg.de>  
 Thomas Woehner <Thomas.Woehner@eah-jena.de>  
 Ralf Peters <ralf.peters@wiwi.uni-halle.de>

---

compareToFIMarket	<i>Compares the welfare of the WoM campaign to that of a fully informed market</i>
-------------------	--

---

### Description

Compares the welfare of the WoM campaign to the welfare of a fully informed market assuming a uniformly distributed willingness to pay.

### Usage

```
compareToFIMarket(campaign)
```

### Arguments

campaign      Word-of-Mouth campaign as instance of class WoMCampaign.

### Value

Data frame containing the profit-maximizing price, the resulting demand, profit, consumer surplus and economic welfare for a fully informed market and a WoM market.

### Author(s)

Michael Scholz <michael.scholz@th-deg.de>

Thomas Woehner <Thomas.Woehner@eah-jena.de>

Ralf Peters <ralf.peters@wiwi.uni-halle.de>

### See Also

[computeOptimalPrice](#) [computeProfit](#) [computeConsumerSurplus](#)

### Examples

```
network <- new("WoMNetwork", size = 1000, avgConnections = 5)
campaign <- new("WoMCampaign", network = network, seedingSize = 10, forwardProbability = 0.2)
comparison <- compareToFIMarket(campaign)
print(comparison)
```

computeConsumerSurplus

*Computes the expected cumulative consumer surplus*

---

### Description

Computes the expected cumulative consumer surplus for a given Word-of-Mouth campaign at a given price.

### Usage

```
computeConsumerSurplus(campaign, price)
```

### Arguments

campaign	Word-of-Mouth campaign as instance of class WoMCampaign.
price	Price as number in [0; 1] where 0 is the minimal and 1 is the maximal price.

### Value

Expected cumulative consumer surplus.

### Author(s)

Michael Scholz <michael.scholz@th-deg.de>  
Thomas Woehner <Thomas.Woehner@eah-jena.de>  
Ralf Peters <ralf.peters@wiwi.uni-halle.de>

### See Also

[computeDemand](#) [computeProfit](#) [computeOptimalPrice](#)

### Examples

```
network <- new("WoMNetwork", size = 1000, avgConnections = 5)
campaign <- new("WoMCampaign", network = network, seedingSize = 10, forwardProbability = 0.2)
surplus <- computeConsumerSurplus(campaign, price = 0.5)
print(surplus)
```

---

computeDemand	<i>Computes the expected demand</i>
---------------	-------------------------------------

---

**Description**

Computes the expected demand for a given Word-of-Mouth campaign at a given price.

**Usage**

```
computeDemand(campaign, price)
```

**Arguments**

campaign	Word-of-Mouth campaign as instance of class WoMCampaign.
price	Price as number in [0; 1] where 0 is the minimal and 1 is the maximal price.

**Value**

Expected demand in number of persons.

**Author(s)**

Michael Scholz <michael.scholz@th-deg.de>  
Thomas Woehner <Thomas.Woehner@eah-jena.de>  
Ralf Peters <ralf.peters@wiwi.uni-halle.de>

**See Also**

[computeRoundDemand](#) [computeProfit](#) [computeConsumerSurplus](#) [computeOptimalPrice](#)

**Examples**

```
network <- new("WoMNetwork", size = 1000, avgConnections = 5)
campaign <- new("WoMCampaign", network = network, seedingSize = 10, forwardProbability = 0.2)
demand <- computeDemand(campaign, price = 0.5)
print(demand)
```

---

`computeInformationCostsThreshold`*Computes the information costs threshold*

---

### Description

Computes the information costs that need to be surpassed in order to generate a higher profit than in a transparent market.

### Usage

```
computeInformationCostsThreshold(campaign)
```

### Arguments

`campaign`      Word-of-Mouth campaign as instance of class `WoMCampaign`.

### Value

Information costs in  $[0; 1]$  that need to be surpassed in order to generate a higher profit than in a transparent market.

### Author(s)

Michael Scholz <michael.scholz@th-deg.de>

Thomas Woehner <Thomas.Woehner@eah-jena.de>

Ralf Peters <ralf.peters@wiwi.uni-halle.de>

### See Also

[computeOptimalPrice](#) [computeProfit](#)

### Examples

```
network <- new("WoMNetwork", size = 1000, avgConnections = 5)
campaign <- new("WoMCampaign", network = network, seedingSize = 10, forwardProbability = 0.2)
threshold <- computeInformationCostsThreshold(campaign)
print(threshold)
```

---

computeOptimalPrice    *Computes the profit-maximizing price*

---

**Description**

Computes the profit-maximizing for a given Word-of-Mouth campaign.

**Usage**

```
computeOptimalPrice(campaign)
```

**Arguments**

campaign            Word-of-Mouth campaign as instance of class WoMCampaign.

**Value**

Profit-maximizing price in [0; 1] where 0 is the lowest possible and 1 is the highest possible price.

**Author(s)**

Michael Scholz <michael.scholz@th-deg.de>

Thomas Woehner <Thomas.Woehner@eah-jena.de>

Ralf Peters <ralf.peters@wiwi.uni-halle.de>

**See Also**

[computeDemand](#) [computeProfit](#) [computeConsumerSurplus](#)

**Examples**

```
network <- new("WoMNetwork", size = 1000, avgConnections = 5)
campaign <- new("WoMCampaign", network = network, seedingSize = 10, forwardProbability = 0.2)
price <- computeOptimalPrice(campaign)
profit <- computeProfit(campaign, price)
print(price)
print(profit)
```

---

computeProfit	<i>Computes the expected profit</i>
---------------	-------------------------------------

---

**Description**

Computes the expected profit for a given Word-of-Mouth campaign at a given price.

**Usage**

```
computeProfit(campaign, price)
```

**Arguments**

campaign	Word-of-Mouth campaign as instance of class WoMCampaign.
price	Price as number in [0; 1] where 0 is the minimal and 1 is the maximal price.

**Value**

Expected profit as number of persons times price.

**Author(s)**

Michael Scholz <michael.scholz@th-deg.de>  
Thomas Woehner <Thomas.Woehner@eah-jena.de>  
Ralf Peters <ralf.peters@wiwi.uni-halle.de>

**See Also**

[computeDemand](#) [computeConsumerSurplus](#) [computeOptimalPrice](#)

**Examples**

```
network <- new("WoMNetwork", size = 1000, avgConnections = 5)
campaign <- new("WoMCampaign", network = network, seedingSize = 10, forwardProbability = 0.2)
profit <- computeProfit(campaign, price = 0.5)
print(profit)
```



---

computeRoundDemand      *Computes the expected demand per round*

---

### Description

Computes the expected demand for a given Word-of-Mouth campaign at a given price and a given round or a given round and all previous rounds

### Usage

```
computeRoundDemand(campaign, price, round, previousRounds = TRUE)
```

### Arguments

campaign      Word-of-Mouth campaign as instance of class WoMCampaign.  
price      Price as number in [0; 1] where 0 is the minimal and 1 is the maximal price.  
round      Round at which or until which the demand per round will be computed.  
previousRounds      Should the demand of all previous rounds be returned or not. Default is TRUE.

### Value

Expected demand in number of persons. Note that the first value in the demand vector is the number of initial consumers when previousRounds is TRUE. The number of initial consumers is  $(1-p)*seedingSize$ .

### Author(s)

Michael Scholz <michael.scholz@th-deg.de>  
Thomas Woehner <Thomas.Woehner@eah-jena.de>  
Ralf Peters <ralf.peters@wiwi.uni-halle.de>

### See Also

[computeDemand](#) [computeProfit](#) [computeConsumerSurplus](#) [computeOptimalPrice](#)

### Examples

```
network <- new("WoMNetwork", size = 1000, avgConnections = 5)
campaign <- new("WoMCampaign", network = network, seedingSize = 10, forwardProbability = 0.2)
demand <- computeRoundDemand(campaign, price = 0.5, round = 3)
print(demand)
```

---

computeWoMIntensity    *Computes the WoM intensity*

---

**Description**

Computes the WoM intensity in a given Word-of-Mouth campaign.

**Usage**

```
computeWoMIntensity(campaign)
```

**Arguments**

campaign            Word-of-Mouth campaign as instance of class WoMCampaign.

**Value**

WoM intensity in [0; 1].

**Author(s)**

Michael Scholz <michael.scholz@th-deg.de>

Thomas Woehner <Thomas.Woehner@eah-jena.de>

Ralf Peters <ralf.peters@wiwi.uni-halle.de>

**Examples**

```
network <- new("WoMNetwork", size = 1000, avgConnections = 5)
campaign <- new("WoMCampaign", network = network, seedingSize = 10, forwardProbability = 0.2)
intensity <- computeWoMIntensity(campaign)
print(intensity)
```

---

show, WoMCampaign-method  
*Shows a WoMCampaign object*

---

**Description**

Shows a WoMCampaign object

**Usage**

```
## S4 method for signature 'WoMCampaign'
show(object)
```

### Arguments

object            An instance of the WoMCampaign-class

### Methods

**list("signature(object = \"WoMCampaign\")")** Shows an WoMCampaign object.

### Author(s)

Michael Scholz <michael.scholz@th-deg.de>

Thomas Woehner <Thomas.Woehner@eah-jena.de>

Ralf Peters <ralf.peters@wiwi.uni-halle.de>

---

show, WoMNetwork-method

*Shows a WoMNetwork object*

---

### Description

Shows a WoMNetwork object

### Usage

```
## S4 method for signature 'WoMNetwork'  
show(object)
```

### Arguments

object            An instance of the WoMNetwork-class

### Methods

**list("signature(object = \"WoMNetwork\")")** Shows an WoMNetwork object.

### Author(s)

Michael Scholz <michael.scholz@th-deg.de>

Thomas Woehner <Thomas.Woehner@eah-jena.de>

Ralf Peters <ralf.peters@wiwi.uni-halle.de>

---

WoMCampaign-class      *Class* WoMCampaign

---

### Description

This class represents a WoM campaign that is performed on a given network to promote a durable good with no variable costs.

### Slots

`network` (WoMNetwork) The network to which the WoM campaign is applied.  
`seedingSize` (numeric) Number of consumers who are initially informed about the good by the firm.  
`forwardProbability` (numeric) Probability at which a consumer forwards information about the good to others.  
`informationCosts` (numeric) Costs to information one consumer about the good.

### Objects from the Class

Objects can be created by calls of the form `new("WoMCampaign", ...)`. This S4 class describes WoMNetwork objects.

### Author(s)

Michael Scholz <michael.scholz@th-deg.de>  
 Thomas Woehner <Thomas.Woehner@eah-jena.de>  
 Ralf Peters <ralf.peters@wiwi.uni-halle.de>

### Examples

```
network <- new("WoMNetwork", size = 1000, avgConnections = 5)
campaign <- new("WoMCampaign", network = network, seedingSize = 10, forwardProbability = 0.2)
print(campaign)
```

---

WoMNetwork-class      *Class* WoMNetwork

---

### Description

This class represents an average random graph.

### Slots

`size` (numeric) The number of consumers in the network.  
`avgConnections` (numeric) Average number of connections per consumer.

### **Objects from the Class**

Objects can be created by calls of the form `new("WoMNetwork", ...)`. This S4 class describes WoMNetwork objects.

### **Author(s)**

Michael Scholz <michael.scholz@th-deg.de>

Thomas Woehner <Thomas.Woehner@eah-jena.de>

Ralf Peters <ralf.peters@wiwi.uni-halle.de>

### **Examples**

```
network <- new("WoMNetwork", size = 1000, avgConnections = 5)
print(network)
```

# Index

## \* **Word-of-Mouth**

WordOfMouth-package, [2](#)

## \* **classes**

WoMCampaign-class, [12](#)

WoMNetwork-class, [12](#)

## \* **methods**

show, WoMCampaign-method, [10](#)

show, WoMNetwork-method, [11](#)

[compareToFIMarket, 3](#)

[computeConsumerSurplus, 3, 4, 5, 7–9](#)

[computeDemand, 4, 5, 7–9](#)

[computeInformationCostsThreshold, 6](#)

[computeOptimalPrice, 3–6, 7, 8, 9](#)

[computeProfit, 3–7, 8, 9](#)

[computeRoundDemand, 5, 9](#)

[computeWoMIntensity, 10](#)

[show, WoMCampaign-method, 10](#)

[show, WoMNetwork-method, 11](#)

[WoMCampaign-class, 12](#)

[WoMNetwork-class, 12](#)

[WordOfMouth \(WordOfMouth-package\), 2](#)

[WordOfMouth-package, 2](#)